

# Andrew Cooke

## Personal Details

**Address:** Luis Thayer Ojeda 596 Dept 304,  
Providencia,  
Santiago,  
Chile.

**Phone:** (56) 9 66038070

**Email:** andrew@acooke.org

**Web:** <http://www.acooke.org>  
<https://github.com/andrewcooke>

**Place of birth:** Harrogate, Yorkshire, UK.

**Nationality:** Chilean / British.

**Languages:** English; Spanish.

## Summary

PhD in astronomy + over 25 years of experience in software engineering. I have written numerical software in Python, Java, C, and Fortran; maintained messaging libraries; designed database schemas and implemented the associated loaders; implemented scalable, responsive web sites; used containerization locally in docker and remotely on Azure; deployed complex systems using ansible and AWS CDK. I am very self-motivated, reliable, independent and productive with many years' experience in telecommuting.

### Professional Interests

– How to involve the client in “lightweight” development — balancing iterative, adaptable development with clear estimates and a useful development history. – Domain-specific and ‘little’ languages/parsing/code-as-data/flexible configuration. These ideas often provide a good abstraction layer for building adaptable, maintainable systems.

– Efficient numerical and semi-numerical algorithms. For example, I have developed new, efficient approaches for filtering data in one and two dimensions; I worked on generating correlated, uniformly distributed random numbers.

### Skills

– Decades of OO design experience.

– Strong mathematical and statistical background.

– Experience with Agile, Requirements-Driven and Iconix (UML) processes.

– Self-motivated problem solver.

– Educated to PhD level (Astronomy, Cambridge University).

**Languages:** Python, C, Java, Javascript, SQL (some OpenCL, OpenGL, Julia).

**Platforms:** Linux (OpenSuse, CentOS, Rocky, Debian), Windows.

**Web:** React, Javascript, Django, Flask, Spring, SVG.

**Databases:** PostgreSQL, PostGIS, Oracle, MySQL; JDBC, SQLAlchemy, Spring.

**Virtualisation:** AWS, Azure, Docker, Kubernetes, VirtualBox.

## Work Experience

### 2008— Senior Software Engineer. ISTI, USA (Telecommute).

- ISTI develop custom software for the geophysical research community; they are based in the USA but have engineers in several countries. On most projects below I was responsible for design and implementation, and usually interacted with the client (typically in parallel with one of the company founders).
- AWS Build and deploy scalable test system using CDK and Ansible (complex system tests running in parallel on many machines).
- Scalability Server-side support for high-availability web application. Hosted on Azure with many components running in Kubernetes, updates pushed via SignalR, data stored in PostgreSQL.
- ETL / Schema Developed PostgreSQL/PostGIS schema and extract, transform, load tools (Python) to store data from earthquake alert tests for further analysis.
- CI / Testing Configured Jenkins with git and JIRA (including a Jenkins plugin to automatically raise and close JIRA issues). Advised on use and helped develop tests.
- Hardware / Numerical A set of loosely coupled C programs that calibrate seismic detectors. These can be run separately, by hand, or under the control of a scheduler for automated calibration.
- GPU / Numerical Optimization of numerical Matlab/Octave code using OpenCL. Reduced calculation from 12m (Xeon CPU) to 10s (low cost NVidia GPU), shifting work from “batch processing” to “interactive data exploration”.
- Web / Database Several projects constructing database representations of complex systems and then providing a variety of ways to access and manipulate that data — both directly (HTML, Ajax) and via additional services (REST, XMLRPC). Implemented with Java (Spring/JSP) and Python (Django/YUI).
- Client Application Python (WXWidgets) GUI to simplify management of remote data processing system, including a “map” of interconnected components (auto-layout via simulated annealing).
- Other I have helped introduce a variety of ideas to the company, including the use of continuous integration, lightweight progress tracking, and wider test use.

### 2007—2008 Software Engineer. MuleSource, San Francisco, USA (Telecommute).

- MuleSource was the company formed to support and develop Mule, an open source Enterprise Service Bus (ESB). I was part of a geographically-disperse team maintaining the core system, particularly TCP related transports.
- XML Schema I was also responsible for the main user-visible change in Mule 2.0: an XML-based configuration system using Spring’s extensible schema.

### 2003—2007 Scientific Programmer. CTIO, La Serena, Chile.

- The team in La Serena was part of a larger development group based in Tucson, USA, that developed software for NOAO observatories.
- Numerical I designed and implemented the Gemini/IRAF GNIRS Package, to process spectral data. This was based on the existing (but incomplete) NIRI package and implemented in IRAF CL/SPP (Fortran).
- SOA / ESB The NOAO Science Archive was developed to store and retrieve astronomical data. I worked on analysis, design, implementation, testing and documentation of the system. This included assessing ESB systems and selecting Mule as a solution that provided good scalability, wide compatibility with existing transports, and support for rapid development with Java-based messages — a good, future-proof balance for a SOA that was still largely internal.

## Work Experience (cont.)

- 2002—2003**    **Head of development / Consultant. Webtron Finance, Santiago, Chile.**  
At Webtron I implemented a system to receive and process financial data. That involved learning, over 7 months, how to develop J2EE-based web applications, in a new language and culture. I started as a single Java programmer, writing to a dictated design, but finished leading a small team (two programmers and web designer) to beat an impossible deadline with shifting requirements.
- 1998—2001**    **Software Engineer. Intertrader Ltd, Edinburgh / Leicester, UK (Telecommute).**  
For the Intertrader CashBox System I designed and implemented most of the server-side application, combining standard Java components (to become ‘J2EE’) within a dynamically configurable framework (similar to the ‘Spring’ framework, although I was unaware of that at the time) to give the flexibility necessary when working for different clients with conflicting requirements.
- 1997—1998**    **Software Engineer. Concept Systems, Edinburgh, UK.**  
Responsible for algorithms to calculate the position of long (5km) cables towed behind boats prospecting for oil. I developed a novel, fast algorithm for median filtering (using a sorted tree for the data within the window) and helped start an internal discussion group to encourage movement from C to C++.
- 1995—1997**    **Postdoc. Institute for Astronomy, Edinburgh, UK.**  
Numerical analysis (Fortran 77; maximum likelihood estimates; integration; optimization) of the distribution of Lyman- $\alpha$  absorption lines to estimate the evolution of the ionizing background at high redshifts.
- 1994**    **Research Assistant. CTIO, La Serena, Chile.**  
Analysis of Hubble and ground-based long-slit and Fabry-Perot observations. Fortran (fitting models of gas flow to 3D spectral data) and IRAF.
- 1988—1993**    **PhD in Astronomy. Institute of Astronomy, Cambridge, UK.**  
Voigt profile fitting in Fortran. Observed (mainly echelle spectroscopy) at AAT, CTIO 4m, WHT. Wrote software in Fortran with IRAF / Imfort to do optimal data extraction (not supported in IRAF for echelle spectra at the time) with automatic cosmic ray rejection.
- 1985—1988**    **BA in Natural Sciences (Maths and Physics); Christ’s College, Cambridge, UK.**  
First class honours (final result and all intermediate examinations); received various scholarships.

## Personal Work

- Plotter**    An extremely basic ‘1D’ plotter that uses a pen on a rotating (and slowly translating) cylinder, controlled by Arduino (C++), driven via serial comms from Processing (Java). I acquired the electronic hardware as mentor in a project encouraging Chilean schoolgirls to be engineers (Niñas Pro 2023).
- Choochoo**    A ‘training diary’ roughly similar to Strava. Extract data from FIT files (as recorded by Garmin sports watches), store them in a PostgreSQL/PostGIS database, and present statistics via a React web interface. Deployment via Docker.
- Previously**    Recursive descent parser for Python; CRC library in Julia; Google App Engine service to generate ‘user icons’; a ‘concatenative’ language; Haskell library for functional images.